ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Design and Implementation of Floating-Point Butterfly Architecture Based on Multi-Operand Adders

K.Lakshmi Narasamma
VLSI & ES,
Department of ECE,
Pace institute of Technology and Sciences,
NH-5, Near Valluramma Temple, Ongole,
Prakasam District, A. P.

K.Sundeep
Associate Professor,
Department of ECE - Communication Systems,
Pace institute of Technology and Sciences,
NH-5, Near Valluramma Temple, Ongole,
Prakasam District, A. P.

Abstract:
In this paper we have here in the processor FFT and FFT butterfly structure, reading, writing and execution addresses. Fast Fourier Transform (FFT) coprocessor having a noticeable impact on the performance of communication systems, has been a hot topic of research for many years. FFT function over the complex numbers in a row, also known as butterfly units have to add and multiply. , FFT architectures applicable to floating point (FP) arithmetic units have become more popular recently, especially the butterfly. FP is dismissing concerns (eg, scaling and overflow / flow) compute-intensive tasks from the general-purpose processor offloads. However, FP butterfly, the main difficulty is slow as compared with the fixed-point counter. To mitigate the slowdown in high-speed FP FP reveals incentive to the development of the butterfly structure.

This brief product- AB e ± ± CD, signed- binary digit (BSD) on the basis of representation to count, add (FDPA) unit to a faster one using drops were invented FP FP proposed butterfly unit. Three method FP FP BSD BSD adder and multiplier stable unit are part of the proposed FDPA. BSD is a carry-limited so as to improve the speed of the unit adder FDPA proposed and three horizontal BSD method adder and multiplier are used. Moreover, the change in the booth is used to speed up the encoding module is BSD. FP results in the synthesis of the proposed structure of the butterfly, but more than in previous images show that the cost is much faster.

The index of signed binary digit (BSD) represented by the butterfly unit, the complex number system, fast Fourier transform (FFT), floating point (FP), the unnecessary number system, in addition to the three method.

INTRODUCTION:

Joseph Fourier in 1811 in a paper presented to the French Academy of Sciences. As soon as the paper is published, the Fourier effect and un- benownst, yet many problem domains, which would have a lasting impact on a range. Fourier presented by the magazine in its original domain (eg, time) equally distributed samples to the frequency domain as a method of transforming a set _nite Fourier transform concept was introduced. Apply the mathematical concept of a set of complex numbers and Fourier in_nite the transition does not involve integration. This is a continuing type of Fourier transform has many applications in physics and engineering, but it is the form of a discrete Fourier transform of the computer systems that can easily [53] can be implemented. The discrete Fourier transform (DFT) and its applications in computer science in 1965 (FFT) to transform the James Cooley and John Tukey [18] led to the fast Fourier. Divide and conquer using the method, FFT O (N2) O (Nlog (N)) also reduces the computational complexity of the DFT. When such signi_cant computational complexity, FFT Fourier transform many di_cult problems were a convenient solution. Turning around the areas of digital signal pro cessing and the number of FFT knots, which in the 20th century, [22] is considered to be one of the most inuential algorithms.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

It is used in many applica- tions and systems, FFT modern computers [22] is one of the most commonly used algorithms. Modern computers allow us to solve the problems of larger and more complex, the past several decades have become significantly faster. However, some large-scale problems still di_culties. Also a (Nlog (N)) FFT with plexity of the realization, some large-scale problems that are tying the modern computers. Through the use of parallelization of computational de_ciencies a way to overcome. By using several computers in parallel to solve a problem, we can greatly reduce the amount of computation time. With the growing realization that the prevalence of multiple and multi-core architectures monplace strategy.

The nature of the distribution of FFT algorithm is an obvious candidate for parallelization. 1965 [18] Since the introduction of the FFT algorithm parallelization it is a great subject of research (and is). Fast Fourier transformation (FFT) circuits have several series of multipliers and adders over the complex numbers; That should be a sufficient number to represent the smarter choice. Using a fixed-point arithmetic, floating point FFT is based on the structures of FFT (FP) operations growing until recently, [1], [2]. The main advantage of a fixed-point arithmetic, the introduction of the FP has a wide dynamic range; But the price is reduced. In addition, IEEE-754-2008 standard use [3] FP an FFT coprocessor for arithmetic, general purpose processors allows collaboration. Offloads compute-intensive tasks from the processor and can lead to high performance.

FP activities, their main drawback is slower when compared to fixed-point units. FP arithmetic as a way to speed up the integration of several operations in a single FP unit, and hence save the delay, area, and power consumption, [2]. Using redundant number systems, FP overcoming slowing the propagation of the carry over the intermediate term, there is no known way to work.

LITERATURE SURVEY:

The Sequential Fast Fourier Transform Explained:

The Discrete Fourier Transform is an operation performed on a series of elements to convert the underlying domain (e.g., time) to frequency, or vise-versa. The result has many useful applications and is one of the most widely used algorithms of the 20th and 21st centuries [22]. The typical DFT operation performed on N elements x1; x2:::xN is de_ned as:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-i2\pi k \frac{n}{N}}$$

As a result of the input range of each element of the array X is an additive that requires the cooperation of every element of x. Thus, an N- DFT operation element in the input array O (N2) is. In the case of a large array of applications, compute the DFT of this difficult time. Therefore, DFT-time-complexity of the algorithm, it is desirable to reduce. Cooley and Tukey Fast Fourier [18] algorithm introduced in 1965, to transform, to significantly reduce the complexity of computing discrete Fourier transform and conquer approach is the partition. FFT method, we have no input in the range of a DFT (NlogN) to be able to count. FFT for a very brief introduction to the original paper [18], but its effect is evident as soon as the number _eld of turning around. Sev- eral documents shortly after being published in more detail and more FFT algorithm [28] generalize. However, for many years after its introduction, the development of some of the technology, "Cooley-Tukey algorithm" was held, and the theoretical aspects of the re- search FFT algorithm, or FFT analysis focused on the details of the proposals.

The Cooley-Tukey Algorithm:

Cooley-Tukey algorithm, [18] as introduced, accounting (both additions and multiplications) to calculate the discrete Fourier transform to reduce the need to use a divide and conquer approach. The Cooley-Tukey algo- rithm, outlined here, radix-2 D (decimation-in-time), and for a simple FFT algorithm serves as our base.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

The basic steps of the algorithm:

1. decimate - two (ie, source 2) of DFT to create small, even or odd set the split of the original input.
2. Multiply - Multiply each element of the source of the unity of the coalition (called twiddle factors [28]).
3. Butterfly - (see picture 1), the other with a small element of the corre- sponding add each element of each of the DFT.

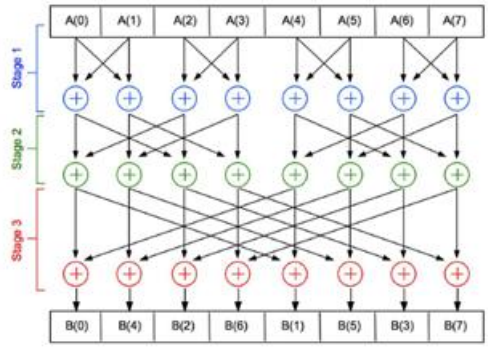

Figure 1: Diagram of the recursive process of a radix-2 DIT Cooley-Tukey FFT algorithm. Example calculates the DFT of an 8-element array in 3 ($log(n)$) recursive stages, with 8 (n) addition and multiplications operations at each stage. Note that the output array (B) is not in the correct order (the indices of the output array elements are jumbled) and must be re-ordered to generate the correct DFT result.

Existing systems

FFT

An FFT processor chip was originally designed. OFDM FFT processor is the focal point of both the transmitter and receiver. FFT high performance, combined with low energy consumption, such as high-throughput approach to the implementation of an ASIC is computationally demanding operation.

Architecture

The FFT and IFFT Equation 2.1 and 2.2 has the property that, if

FFT(Re(xi)+ jIm(xi)) = Re(Xi)+ jIm(Xi)

and

IFFT(Re(Xi)+ jIm(Xi)) = Re(xi)+ jIm(xi),

where xi and Xi are N words long sequences of complex valued, samples and sub-carriers respectively, then

1/N * FFT(Im(Xi)+ jRe(Xi)) = Im(xi)+ jRe(xi).

Therefore, it is necessary not only to discuss the implementation of the FFT equalizer.

To calculate the inverse transform, the real and imaginary part of the input and output are changed. N 1 / n scaling a power of two, so that the right binary word log2 (N) bits in the same switch. Even simpler, binary point left log2 (N) bits is shifted to remember. If ever, did not show up to change a bit, depending on how it is used, the output from IFFT, is required.

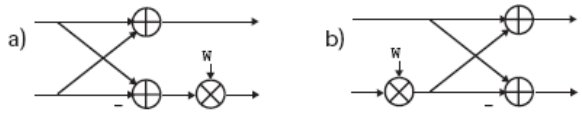

Figure 4.1: A radix-2 DIF butterfly (a) and a radix-2 DIT butterfly (b), where W is the twiddle factor.

FFT algorithm is the basic building block [23] can be realized with a butterfly operation. Frequency (DIF) time, (d) and the decimation of the butterfly in the death of two types of operations, the two are shown in Figure 4.1. DIF is the difference between the before and after the addition or subtraction and multiplication of featured twiddle factor is in place. FFT based on the division to conquer and due to the input range is N R source, known as the point length N = RP, so, and p positive integer is the most effective. An N-point FFT, to count the butterflies are connected to the p stages.

The map is an example of a hardware-point radix-made N = 16 -2 DIF FFT is shown in Figure 4.2. The input data, x (N), the output of data occurs in a random order, however, X (N) to return to observe it. Reversed the order of the data generated to re-order bit is known and described in Section 5.3. Figure 4.2 FFT bit reversed order, which will result in the reshaping of the input and output, so it is possible to form the natural order. Figure 4.2, moving vertically, parallel data paths and reconfigure the product to the natural order, the way to control cross connections while, and think. As well as all the arrows in Figure 4.2, if one turned around from the FFT, DIF FFT is performed instead.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

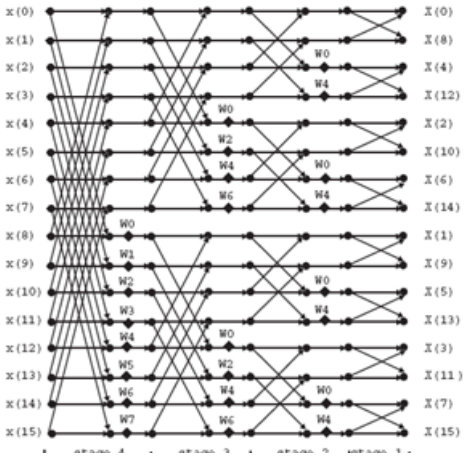

Figure 4.2: A hardware mapped N = 16-point radix-2 DIF FFT algorithm.

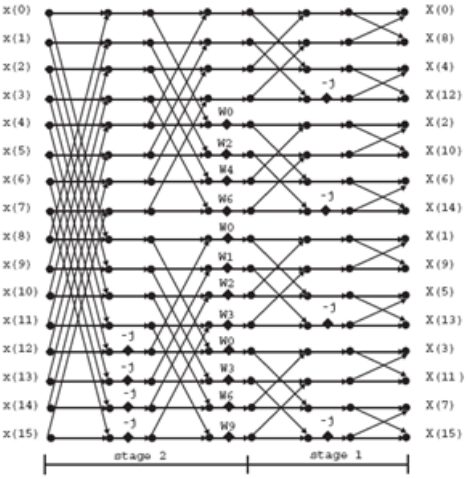

Figure 4.3: A hardware mapped N = 16-point radix-22 DIF FFT algorithm.

PROPOSED SYSTEMS:

Proposed Butterfly Architecture:

FFT efficient algorithms [5] in which the FFT calculation of input N- (n / 2) FFT -input simplified calculations based on the hardware could not run. Continuing the distribution of 2-input FFT block, also known as the lead unit of the butterfly. Multiply- were in fact proposed the butterfly unit is a complex FP Combine with operands. Expanding complex numbers, figure. 1 shows the required modules. According to Fig. 1, the constituent operations for butterfly unit are a dot-product (e.g., BreWim+ BimWre) followed by an addition/subtraction which leads to the proposed FDPA operation (e.g., BreWim+ BimWre+ Aim). Implementation details of FDPA, over FP operands, are discussed below.

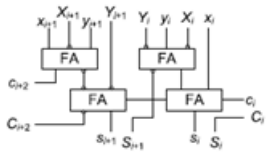

Fig. 2. BSD adder (two-digit slice).

TABLE I
GENERATION OF A PP

$W_{i+1}^- W_{i+1}^+$	$W_i^- W_i^+$	$\|W_{i+1}^- W_{i+1}^+ W_i^- W_i^+\|$	PP_i
0 0	0 0	0	0
0 0	0 1	1	B
0 0	1 1	−1	$-B$
0 1	0 0	2	$2 \times B$
1 1	0 0	−2	$-2 \times B$

Or, objective, DRE, and Bim's significands BSD, however, there are exponents of all inputs, predicted (after removal of non-discrimination), the two's complement representation. Within each of these represent the position of the binary values {-1, 0, 1} is a bit of a negative (negabit) and a bit of a positive (posibit) represented by. BSD carry a limited additional wiring for the numbers shown in Fig. 2, capital (short) refers to the characters negabits (posibits). The adder critical path delay will have three full-adders. After three FP adder enclosed FDPA the proposed method is a recurring FP multiplier.

A. Proposed vast floating point multiplier:

The proposed multiplier, as well as other parallel multipliers, two major steps vconsists, namely, a partial product generation (PPG) vand PP reduction (PPR). However, in contrast to the traditional multipliers, our module production in repetitive format is required for the final carry propagating adder keeps andvhence. Exponents of the input operands is done in the traditional FP multipliers are taken care of in the same way; However, normalization and rounding butterfly Architecture (ie, three method adder) are left to the next block.

1) partial product generation: PPG foot because of the proposed multiplier input operands (B, W, B_, W_) representation is completely different from the

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

traditional one. Moreover, Wre, Wim [5] constants, given that the multiplications in Fig. 1 (significands over) are calculated through a series of shifters and adders. With the intention of reducing the number of adders, we modified Booth encoding W, significand storage [4]. Wre, Wim, representing a PP mutant booth, multiplicand given the choice from A to B, as shown in Table I. Fig, the coefficient W of each of two binary positions. Shows the wiring needed for the production of 3 ppibased on Table I where each PP consists of (n + 1) digits (i.e., binary positions).

2) Partial Product Reduction: The major constituent of the PPR step is the proposed carry-limited addition over the operands represented in BSD format. This carry-limited addition circuitry is shown in Fig. 2 (two-digit slice). Since each PP (PPi)is (n + 1)-digit (n, . . . , 0) which is either B (n − 1, . . . , 0) or 2B (n, . . . , 1), the length of the final product maybe more than 2n.

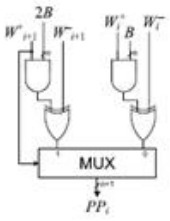

Fig. 3. Generation of the ith PP.

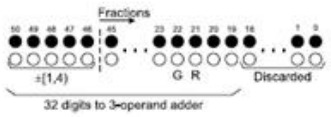

Fig. 4. Digits to three-operand adder.

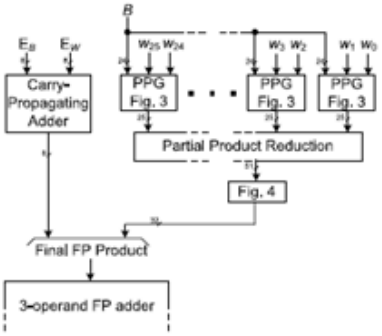

Fig. 5. Proposed redundant FP multiplier.

The inputs A and B (24 bits) can be found in embedded Assuming significands BSD; W The modified Booth encoding (25 bits) are represented in the last PP, 24- (binary position) Width (instead of 25), W has become the most significant bit is always 1. pps four levels, given that the full adders 12 BSD. ?? In B} Given that [1, 2) and [1, 2), in the final product ?? There is} W [1, 4) and 48 binary position (47 inadequate ... 0). Consequently, there are fractions of 0 down to 45 positions. Similar to the standard binary representation, Guard (G) and round (R) locations are sufficient to correct rounding. Therefore, the final product of binary positions in the last part of the only 23 + 2 with an error <2-23 are required to ensure. Semi-finished product of 46 points to 25 from 0 to 20 in binary positions are out of choosing locations, however, the next step is to cause the addition of carries from the G and R positions. However, because of the limited carry BSD addition, in contrast to the standard binary addition, the 20 and 19 positions as the cause carries. Overall, 18 to 0, respectively, the positions of the final product is not useful, and hence it is possible to have a flexible PPR tree. Fig. 4 shows three digits to the method passed to the adder. Fig. 5 shows the proposed recurrent FP multiplier. Three vast floating point adder B of the proposed method FP method to handle the addition of three more delays in the process, power, and which could lead to the use of the area is to concatenate two FP adders. FP three adders could be a good way to use the method [6], [7].

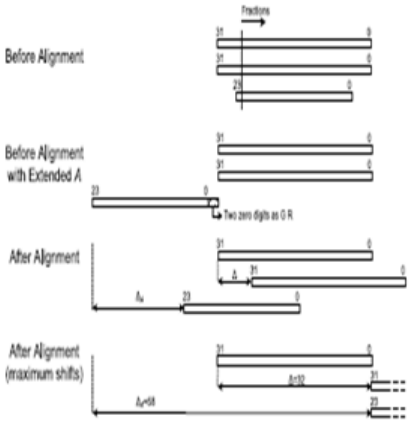

Fig. 6. Proposed three-operand alignment scheme.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

CONCLUSION:

In this paper we have here in the processor FFT and FFT butterfly structure, reading, writing and execution addresses. But the high-risk area, which is faster than the previous works, we have a high-speed FP butterfly structure, proposed. The reason for this is the speed of development is twofold: it eliminates the carry-propagation significands 1) BSD represented, and 2) the abbreviation of the proposed new FDPA unit. FP butterfly multiplications and additions need to be combined with the unit; Thus, high-speed additional LZD, normalize, remove, and can be achieved by rounding units. The next research FP adder three dual-line method applied to the structure and other unnecessary FP FP representations may be planning on using. Moreover, the design stage of the abolition of the use of improved techniques (ie, repeating LZD, normalize, and rounding) of the estimated costs, however, led to faster architectures.

REFERENCES:

[1] E. E. Swartzlander, Jr., and H. H. Saleh, "FFT implementation withfused floating-point operations," IEEE Trans. Comput., vol. 61, no. 2,pp. 284–288, Feb. 2012.

[2] J. Sohn and E. E. Swartzlander, Jr., "Improved architectures for afloating-point fused dot product unit," in Proc. IEEE 21st Symp. Comput.Arithmetic, Apr. 2013, pp. 41–48.

[3] IEEE Standard for Floating-Point Arithmetic, IEEE Standard 754-2008,Aug. 2008, pp. 1–58.

[4] B. Parhami, Computer Arithmetic: Algorithms and Hardware Designs,2nd ed. New York, NY, USA: Oxford Univ. Press, 2010.

[5] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculationof complex Fourier series," Math. Comput., vol. 19, no. 90, pp. 297–301,Apr. 1965.

[6] A. F. Tenca, "Multi-operand floating-point addition," in Proc. 19th IEEESymp. Comput. Arithmetic, Jun. 2009, pp. 161–168.

[7] Y. Tao, G. Deyuan, F. Xiaoya, and R. Xianglong, "Three-operand floating-point adder," in Proc. 12th IEEE Int. Conf. Comput. Inf. Technol.,Oct. 2012, pp. 192–196.

[8] A. M. Nielsen, D. W. Matula, C. N. Lyu, and G. Even, "An IEEE compliant floating-point adder that conforms with the pipeline packet forwarding paradigm," IEEE Trans. Comput., vol. 49, no. 1, pp. 33–47,Jan. 2000.

[9] P. Kornerup, "Correcting the normalization shift of redundant binary representations," IEEE Trans. Comput., vol. 58, no. 10, pp. 1435–1439,Oct. 2009.

[10] 90 nm CMOS090 Design Platform, STMicroelectronics, Geneva, Switzerland, 2007.

[11] J. H. Min, S.-W. Kim, and E. E. Swartzlander, Jr., "A floating-point fused FFT butterfly arithmetic unit with merged multiple-constant multipliers," in Proc. 45th Asilomar Conf. Signals, Syst. Comput., Nov. 2011,pp. 520–524.